Release Notes for the QNX Neutrino 6.4.0 BSP for Kontron nanoETXexpress-SP trunk#

System requirements#

Target system#

- QNX Neutrino RTOS 6.4.0
- Board version: Kontron nanoETXexpress-SP
- EEPROM and burner for IPL

Host development system#

- QNX Momentics 6.4.0
- Terminal emulation program (Qtalk, Momentics IDE Terminal, tip, HyperTerminal, ~minicom, etc.)
- RS-232 serial port
- NULL-modem serial cable
- Ethernet link

Getting Started#

Step 1: Connect your hardware#

1. Connect the serial cable to the terminal debug port of the Kontron nanoETXexpress-SP board and to the first serial port on the host machine (e.g. ser1 on a Neutrino host).

On your host, run your terminal application with the following configuration:

- Baud: 115200
- Bits: 8
- Stop bits: 1
- Parity: none
- Hardware flow control: none

Note:If you have a Neutrino host with a serial mouse, you may have to move the mouse to the second serial port on your host, because some terminal programs require the first serial port.

Step 2: Build the BSP#

You can build a BSP OS image from the source code or the binary components contained in a BSP package.

For instructions about building a BSP OS image, please refer to the chapter Working with a BSP in the Building Embedded Systems manual.

Step 3 - (BIOS Boot): Transfer the OS image to a bootable USB memory stick#

1. On a Neutrino host, plug the USB key in a USB port (the following steps assume the USB key was mounted as /dev/hd1). At a command prompt:
2. fdisk /dev/hd1
3. Delete existing partitions (hit d, then s)
4. Create a new type 79 partition (hit c, then 79, then 0, then numCylinders-1, then s)
5. Make the partition bootable (hit b, then s)

6. Slay and restart devb-umass
7. dinit /dev/hd1t79
8. dloader /dev/hd1 pc1
9. dloader /dev/hd1t79 pc2
10. Slay and restart devb-umass
Note that steps 1-10 only need to be done once.
11. mount -t qnx /dev/hd1t79 /stick
12. cp -V ifs-kontron_nanoSP.bin
13. Ensure the Kontron nanoETXexpress-SP BIOS is configured to boot from the USB stick
14. Plug the USB key in one of the Kontron nanoETXexpress-SP USB ports
15. Restart the Kontron nanoETXexpress-SP

Note that for booting a BIOS based image, the image must be built with the bios option, ie:
`virtual=x86,bios +compress`

Step 3 - (Fastboot)#

1. Ensure your image is small enough to fit in EEPROM with the microcode (images/CMC_kontron.bin) at 0xD0000. If you have an EEPROM chip of 1MB, that means the entire image must fit in the first 0xD0000 (832K). A larger EEPROM chip will allow you to have a larger image, so long as the microcode is located at 0xD0000 and the IPL is at 0xF0000.
2. Go to the images subdirectory of the BSP and run:
3. ./mkrom.sh kontron-nanoSP.bin
Note that the mkrom.sh script was written for a 1MB EEPROM chip, you'll have to change it for larger chips. Its contents are as follows:

```
#!/bin/bash
# This script assumes an EEPROM size of 1MB
# Example usage: ./mkrom.sh ifs-nanoSP.bin

if [ "$1" == "" ]; then
   echo "Must specify the image file"
   exit 1;
fi

#pad it out to 0xD0000
mkrec -s832k -r -ffull $1 > tmp1.bin

#tack on 64k microcode
cat tmp1.bin CMC_kontron.bin > tmp2.bin

#pad image out to 0xF0000
mkrec -s960k -ffull -r tmp2.bin > tmp3.bin

#tack on 64k IPL
cat tmp3.bin ../install/x86/boot/sys/ipl-kontron > fastboot.rom

rm tmp1.bin tmp2.bin tmp3.bin
```

4. This will create a fastboot.rom image that can be written to the EEPROM using an EEPROM burner
5. Boot the Kontron nanoETXexpress-SP with the BIOS disabled
6. The IPL will ask you if you want to scan flash for an image or receive an image over the serial link. 6a. Select "f" to boot the image in EEPPROM
6b. Select "d" to receive an image over the serial link. Then run **`sendnto -d/dev/ser1 -b115200 ifs-at91sam9263ek.bin`** from the host connected to the target.

Note that for booting a Fastboot based image, the image must be built with the nobios option, ie:
`virtual=x86,nobios +compress`

Driver Command Summary#

The driver command lines below are specific to the Kontron nanoETXexpress-SP board. See the online docs for each driver for additional command-line options and other details.

NOTE: Some of these drivers are commented out in the default buildfile. To use the drivers in the target hardware, you'll need to uncomment them in your buildfile, rebuild the image, and load the image into the board.

Component	Buildfile Command	Required Binaries	Required Libraries	Source Location
Startup	`startup-kontron_nanoSP` `startup-bios`	.	.	`src/hardware/startup/boards/kontron_nanoSP`
Serial	`devc-ser8250 -c1843200 -e -F -S -b115200 3f8,4` `devc-ser8250 -c1843200 -e -S -F -b115200 2f8,3`	devc-ser8250	.	`src/hardware/devc/ser8250`
PCI	`pci-poulsbo &` `pci-bios`	pci-poulsbo pci-bios	.	`src/hardware/pci/poulsbo`
Network	`io-pkt-v4-hc -de1000 -ptcpip`	io-pkt-v4-hc ifconfig	devn-e1000.so libsocket.so devnp-shim.so	Binary Only}
USB	`io-usb -duhci -dehci`	io-usb usb	devu-ohci.so devu-ehci.so libusbdi.so	QNX SDP 6.4.x (Binary Only)
SD/MMC	devb-mmcsd	devb-mmcsd	libcam.so io-blk.so cam-disk.so fs-qnx4.so	`src/hardware/devb/mmcsd`
Audio	`io-audio -dintel_hda board_name=nanoETXexpressSP`	io-audio mix_ctl waveout waverec	libasound.so deva-ctrl-intel_hda.so deva-mixer-hda.so	`src/hardware/deva/ctrl/intel_hda`
Graphics	`io-display -d vid=0x8086,did=0x8108`	io-display	devg-poulsbo.so libgf.so.1 libGLES_CM.so.1 libffb.so.2 libm.so.2	`src/hardware/devg/poulsbo`
EIDE	`devb-eide`	devb-eide	libcam.so.2 io-blk.so cam-disk.so fs-qnx4.so fs-dos.so	QNX SDP 6.4.x (Binary Only)

SMbus resource manager	`smb-poulsbo`	smb-poulsbo read_smb write_smb	.	`src/hardware/support/smb-poulsbo-pub`
Touch Screen	`io-hid -d egalax` `devi-hid -r -P touch`	io-hid devi-hid	devh-egalax.so libhiddi.so.1 devh-usb.so	QNX SDP 6.4.x (Binary Only)

Some of the drivers are commented out in the default buildfile. To use the drivers in the target hardware, you'll need to uncomment them in your buildfile, rebuild the image, and load the image into the board.

Additional Graphics Notes#